FORM PTO-1390 (Modified) (REV 11-2000) U.S. DEPARTMENT OF COMMERCE PATENT AND TRADEMARK OFFICE

ATTORNEY'S DOCKET NUMBER: T2147-907330

TRANSMITTAL LETTER TO THE UNITED STATES DESIGNATED/ELECTED OFFICE (DO/EO/US) CONCERNING A FILING UNDER 35 U.S.C. 371

U.S. APPLICATION NO. (IF KNOWN, SEE 37 CFR): 09/889856

INTERNATIONAL APPLICATION NO.	INTERNATIONAL FILING DATE	PRIORITY DATE CLAIMED
PCT/FR00/03230	21 November 2000	23 November 1999

TITLE OF INVENTION:
COMPUTER DEVICE FOR MAKING SECURE MESSAGES AT A NETWORK LAYER

APPLICANT(S) FOR DO/EO/US:
Francois CUNCHON, Rene MARTIN, and Lap TRAN MINH

Applicant herewith submits to the United States Designated/Elected Office (DO/EO/US) the following items and other information:

1. ☒ This is a **FIRST** submission of items concerning a filing under 35 U.S.C. 371.
2. ☐ This is a **SECOND** or **SUBSEQUENT** submission of items concerning a filing under 35 U.S.C. 371.
3. ☒ This is an express request to begin national examination procedures (35 U.S.C. 371(f)). The submission must include itens (5), (6), (9) and (24) indicated below.
4. ☒ The US has been elected by the expiration of 19 months from the priority date (Article 31).
5. ☒ A copy of the International Application as filed (35 U.S.C. 371 (c) (2))
   a. ☐ is attached hereto (required only if not communicated by the International Bureau).
   b. ☒ has been communicated by the International Bureau.
   c. ☐ is not required, as the application was filed in the United States Receiving Office (RO/US).
6. ☒ An English language translation of the International Application as filed (35 U.S.C. 371(c)(2)).
   a. ☒ is attached hereto.
   b. ☐ has been previously submitted under 35 U.S.C. 154(d)(4).
7. ☐ Amendments to the claims of the International Application under PCT Article 19 (35 U.S.C. 371 (c)(3))
   a. ☐ are attached hereto (required only if not communicated by the International Bureau).
   b. ☐ have been communicated by the International Bureau.
   c. ☐ have not been made; however, the time limit for making such amendments has NOT expired.
   d. ☐ have not been made and will not be made.
8. ☐ An English language translation of the amendments to the claims under PCT Article 19 (35 U.S.C. 371(c)(3)).
9. ☒ An oath or declaration of the inventor(s) (35 U.S.C. 371 (c)(4)).
10. ☐ An English language translation of the annexes of the International Preliminary Examination Report under PCT Article 36 (35 U.S.C. 371 (c)(5)).
11. ☐ A copy of the International Preliminary Examination Report (PCT/IPEA/409).
12. ☒ A copy of the International Search Report (PCT/ISA/210).

Items 13 to 20 below concern document(s) or information included:

13. ☒ An Information Disclosure Statement under 37 CFR 1.97 and 1.98.
14. ☒ An assignment document for recording. A separate cover sheet in compliance with 37 CFR 3.28 and 3.31 is included.
15. ☒ A **FIRST** preliminary amendment.
16. ☐ A **SECOND** or **SUBSEQUENT** preliminary amendment.
17. ☐ A substitute specification.
18. ☒ A change of power of attorney and/or address letter.
19. ☐ A computer-readable form of the sequence listing in accordance with PCT Rule 13ter.2 and 35 U.S.C. 1.821 - 1.825.
20. ☐ A second copy of the published international application under 35 U.S.C. 154(d)(4).
21. ☐ A second copy of the English language translation of the international application under 35 U.S.C. 154(d)(4).
22. ☐ Certificate of Mailing by Express Mail
23. ☒ Other items or information:

Verification of Translator; PCT forms: Demande Internationale Publiee, PCT/IB/301, 304 & 308; PCT/RO/101, copies of the IDS references,

JC18 Rec'd PCT/PTO 23 JUL 2001

U.S. APPLICATION NO. (IF KNOWN, SEE 37 CFR	INTERNATIONAL APPLICATION NO.	ATTORNEY'S DOCKET NUMBER
09/889856	PCT/FR00/03230	T2147-907330

24. The following fees are submitted:.

				CALCULATIONS	PTO USE ONLY
BASIC NATIONAL FEE (37 CFR 1.492 (a) (1) - (5)) :					
☐ Neither international preliminary examination fee (37 CFR 1.482) nor international search fee (37 CFR 1.445(a)(2)) paid to USPTO and International Search Report not prepared by the EPO or JPO **$1000.00**					
☒ International preliminary examination fee (37 CFR 1.482) not paid to USPTO but International Search Report prepared by the EPO or JPO **$860.00**					
☐ International preliminary examination fee (37 CFR 1.482) not paid to USPTO but international search fee (37 CFR 1.445(a)(2)) paid to USPTO **$710.00**					
☐ International preliminary examination fee (37 CFR 1.482) paid to USPTO but all claims did not satisfy provisions of PCT Article 33(1)-(4) **$690.00**					
☐ International preliminary examination fee (37 CFR 1.482) paid to USPTO and all claims satisfied provisions of PCT Article 33(1)-(4) **$100.00**					
ENTER APPROPRIATE BASIC FEE AMOUNT =				**$860.00**	
Surcharge of **$130.00** for furnishing the oath or declaration later than ☐ 20 ☐ 30 months from the earliest claimed priority date (37 CFR 1.492 (e)).				**$0.00**	
CLAIMS	NUMBER FILED	NUMBER EXTRA	RATE		
Total claims	6 - 20 =	0	x $18.00	**$0.00**	
Independent claims	3 - 3 =	0	x $80.00	**$0.00**	
Multiple Dependent Claims **(check if applicable).**			☐	**$0.00**	
TOTAL OF ABOVE CALCULATIONS =				**$860.00**	
☐ Applicant claims small entity status. (See 37 CFR 1.27). The fees indicated above are reduced by 1/2.				**$0.00**	
SUBTOTAL =				**$860.00**	
Processing fee of **$130.00** for furnishing the English translation later than ☐ 20 ☐ 30 months from the earliest claimed priority date (37 CFR 1.492 (f)). +				**$0.00**	
TOTAL NATIONAL FEE =				**$860.00**	
Fee for recording the enclosed assignment (37 CFR 1.21(h)). The assignment must be accompanied by an appropriate cover sheet (37 CFR 3.28, 3.31) **(check if applicable).**			☒	**$40.00**	
TOTAL FEES ENCLOSED =				**$900.00**	
				Amount to be: refunded	$
				charged	$

a. ☒ A check in the amount of **$900.00** to cover the above fees is enclosed.

b. ☐ Please charge my Deposit Account No. ______ in the amount of ______ to cover the above fees. A duplicate copy of this sheet is enclosed.

c. ☒ The Commissioner is hereby authorized to charge any additional fees which may be required, or credit any overpayment to Deposit Account No. **501165** A duplicate copy of this sheet is enclosed.

d. ☐ Fees are to be charged to a credit card. **WARNING:** Information on this form may become public. **Credit card information should not be included on this form.** Provide credit card information and authorization on PTO-2038.

NOTE: Where an appropriate time limit under 37 CFR 1.494 or 1.495 has not been met, a petition to revive (37 CFR 1.137(a) or (b)) must be filed and granted to restore the application to pending status.

SEND ALL CORRESPONDENCE TO:

Edward J. Kondracki
MILES & STOCKBRIDGE P.C.
Suite 500, 1751 Pinnacle Drive
McLean, VA 22102-2144

SIGNATURE

Edward J. Kondracki
NAME

20,604
REGISTRATION NUMBER

July 23, 2001
DATE

T2147-907330-US 3876/JMD(PCT)

IN THE UNITED STATES DESIGNATED/ELECTED OFFICE (D.O./E.O./US)

Applicant:	Francois CUNCHON et al.
International Application No.:	PCT/FR00/03230
International Filing Date:	21 November 2000
U.S. Serial No.:	To be Assigned
U.S. Filing Date:	July 23, 2001
For:	**COMPUTER DEVICE FOR MAKING SECURE MESSAGES AT A NETWORK LAYER**

McLean, Virginia

PRELIMINARY AMENDMENT

Honorable Commissioner of Patents
and Trademarks
Washington, D.C. 20231

Sir:

Please amend the subject application, filed concurrently herewith, as indicated below:

IN THE TITLE:

Please cancel the title in its entirety and substitute the following new title:

-- COMPUTING DEVICE FOR SECURING MESSAGES IN A NETWORK LAYER—

Page 1, after the title and before the first paragraph, insert the following title at the left-hand margin:

--FIELD OF THE INVENTION--;

Page 1, at line 6, before the send paragraph, insert the following heading at the left-hand margin:

--<u>DESCRIPTION OF RELATED ART</u>--;

Page 1, at line 30, before the paragraph beginning "The subject of the...", insert the following heading at the left-hand margin:

--<u>SUMMARY OF THE INVENTION</u>--;

Page 2, at line 18, before the paragraph beginning "A description...", insert the following heading:

--<u>BRIEF DESCRIPTION OF THE DRAWINGS</u>--;

Page 3, at line 1, before the first paragraph, insert the following paragraph at the left-hand margin:

--<u>DESCRIPTION OF THE PREFERRED EMBODIMENT(S)</u>--;

Please delete the paragraph on page 3, beginning at line 1 and ending at line 11, in its entirety, and insert the following paragraph. The changes that were made in the paragraph are shown by underlining and bracketing in an attachment to this Preliminary Amendment:

--Referring to Fig. 1, a computing device 67 is physically linked to a first private network 69 and a computing device 68 is physically linked to a second private network 70. Messages can circulate in complete confidentiality through each of the private networks 69 and 70, insofar as no intrusion can be accomplished from outside these networks. However, if the device 67 sends a message to the device 68 using services of a public network 71, confidentiality is not assured without taking particular precautions. The public network 71 is for example the network known as the Internet, often represented in the form of a cloud in literature. The public network 71 comprises several networks 72, 73, interconnected by means of computing devices such as a computing device 65, not controlled by the devices 67, 68.--

Page 13, after line 28, insert the following new paragraph:

--While this invention has been described in conjunction with specific embodiments thereof, it is evident that many alternatives, modifications and variations will be apparent to those skilled in the art. Accordingly, the preferred embodiments of the invention as set forth herein, are intended to be illustrative, not limiting. Various changes may be made without departing from the true spirit and full scope of the invention as set forth herein and defined in the claims.—

<u>IN THE CLAIMS</u>:

Please cancel claims 1 – 5 in their entirety and insert the following new claims:

--6. A computing device (1) comprising a memory (2) and a network security layer (9) for applying a securing operation upon presentation of a message (M1) in the memory (2);

- the network security layer (9) having an initial state (12) adapted to be switched to a first state (25) that saves an execution context (CE) in an area (52) of the memory (2) upon presentation of the message (M1);

- the network security layer having a second state (33) and adapted to be switched to the second state to call a first function (F9) for processing the message (M1), passing as parameters of said first function (F9), at least an address (@F13) of a second function (F13) and a pointer PZS(M1) to the area (52) of the memory (2), the network security layer being switched to its second state (33) upon saving of the execution context (CE);

- the network security layer being immediately switched back to the initial state (12) upon an acknowledgement of the first function (F9) before the processing of the message (M1);

- the network security layer (9) having a third state (56) and adapted to be switched from the initial state (12) to the third state (56) for restoring the execution context (CE) after which the network security layer (9) is switched back to the initial state in response to a jump to the address (@F13) of the second function.

7. A computing device (1) according to claim 6, further comprising several chained pointers PZS(M1), PZS(M'1) adapted to be restored at the time of the jump to said address (@F13).

8. A computing device (1) according to claim 6, wherein the call of the first function (F9) passes as a parameter a correlation variable (VC1) restored at the time of the jump to the address (@F13).

9. A computing device (1) according to claim 7, wherein the call of the first function (F9) passes as a parameter a correlation variable (VC1) restored at the time of the jump to the address (@F13).

10. A method for creating code for a fast network security layer (9) from the code of a standard network security layer in a kernel layer (6) of a computing device (1), comprising:

- a first step for modifying, in the code of said standard network security layer, a first code sequence adapted to be activated by the presentation of a message to which a securing operation is to be applied, by inserting into the first sequence, before calling a first securing function (F1), a second code sequence,

- beginning the second code sequence by saving a current execution context (CE) when the first sequence is executed, and

- making a call to a second securing function (F9),

-ending the second code sequence with a first jump to the end of the first code sequence;

- a second step for generating a third code sequence of a third function (F13) by copying said first modified code sequence, and then inserting said third code sequence into said first modified code sequence, and

- restoring the saved execution context (CE) by a fourth code

sequence after a call to the first function (F1) with a second jump to said fourth code sequence at the start of the third sequence.

11. A method for obtaining a first secure message from a second message, by means of a computing device (1) comprising a network security layer (9) to which said second message is presented, characterized in that it comprises:

- saving an execution context of the network security layer after the presentation of said second message;

- sending a request for a securing operation by the network security layer to an element outside the network security layer;

- immediately acknowledging by said external element said request so as to place the network security layer in an initial state that does not use any resources of the computing device (1); and

- presenting the message secured by the securing operation that results from said request to activate a restoration of the saved execution context in the network security layer by said external element.--

IN THE ABSTRACT:

Please delete the Abstract at page 16 in its entirety and substitute the following new Abstract.

-- **ABSTRACT**

A computing device (1) comprising a memory (2) and a network security layer (9) for applying a securing operation upon presentation of a message (M1) in the memory (2) is characterized in that:

- the presentation of the message (M1) switches the network security layer (9) from an initial state (12) to a first state (25) that saves an execution context (CE) in an area (52) of the memory (2);

- the saving of the execution context (CE) switches the network security layer from the first state (25) to a second state (33) that calls a first function (F9) for processing the message (M1), passing as parameters of said first function (F9) at least an address (@F13) of said function (F13) and a pointer PZS(M1) to the area (52) of the memory (2);

- immediately switching the network security layer back to the initial state (12) upon an acknowledgement of the first function (F9), before the processing of the message (M1), and;

- switching the network security layer (9) from the initial state (12) to a third state (56) that restores the execution context (CE) before switching the network security layer (9) back to the initial state upon a jump to the address (@F13) of a second function.--

REMARKS

This Preliminary Amendment is filed to insert headings to conform the application to U.S. practice, to eliminate the use of multiple dependent claims, and to correct informalities in the specification, claims and abstract resulting from a literal translation of the French text.

Early action on the merits is earnestly solicited.

Respectfully submitted,

MILES & STOCKBRIDGE P.C.

Date: July 23, 2001

By: _____
Edward J. Kondracki
Registration No. 20,604

1751 Pinnacle Drive – Suite 500
McLean, VA 22102-3833
Tel.: 703/903-9000
Fax: 703/610-8686

Paragraph on page 3, beginning at line 1 and ending at line 11, showing changes that were made by underlining and bracketing:

--Referring to Fig. 1, a computing device 67 is physically linked to a first private network 69 and a computing device 68 is physically linked to a second private network 70. Messages can circulate in complete confidentiality through each of the private networks 69 and 70, insofar as no intrusion can be accomplished from outside these networks. However, if the device 67 sends [an] a message to the device 68 using services of a public network 71, confidentiality is not assured without taking particular precautions. The public network 71 is for example the network known as the Internet, often represented in the form of a cloud in [the] literature. The public network 71 comprises several networks 72, 73, interconnected by means of computing devices such as a computing device 65, not controlled by the devices 67, 68.--

COMPUTER DEVICE FOR MAKING SECURE MESSAGES AT A NETWORK LAYER

The field of the invention is that of computer networks, and more particularly that of securing the routing of messages in these networks.

A public network like the Internet makes it possible to interconnect many private networks linked by access points and routers that route the messages. Ease of access to such a network is an advantage for the free flow of ideas and information, but it is also a disadvantage for the confidentiality of certain information. That is why it is necessary to secure certain messages so that the recipient alone can understand them, and can be sure of their origins and/or their integrity.

A message securing operation is possible in various communication layers of a computing device. For example, in a user layer, an application such as http, ftp or mail can be responsible for performing encryption and decryption, and signature and authentication operations. Generally, the message is only available in the user layer of the initial sender and the final recipient.

According to the prior art, it is possible to provide for the securing operation to be performed in a network layer, wherein a network security layer such as Ipsec handles the securing operation at the very level as the routing of the messages. This makes it possible to create virtual private networks, which use the resources of the public network by means of a known tunnel effect. The network layer is generally considered to be a communication resource of a computing device. The implementation of the network security later resulting from this consideration in the kernel layer of an operating system of the computing device relieves the user layer of the securing operations.

However, some securing operations are long because they apply numerous calculations to the content of a message to be secured. The operating system's wait for the return of a function that gives the result of the operation has the disadvantage of inhibiting the computing device.

The subject of the invention is a computing device comprising a memory and a network security layer for applying a securing operation upon presentation of a message in the memory. In order to eliminate the disadvantage mentioned above, the computing device is characterized in that:

- the presentation of the message switches the network security layer from an initial state to a first state that saves an execution context in an area of the memory;

- the saving of the execution context switches the network security layer from the first state to a second state that calls a first function for processing the message, passing as parameters of said first function at least an address of a second function and a pointer to the area of the memory;

- an acknowledgement of the first function before the processing of the message immediately switches the network security layer back to the initial state;

- a jump to the address of the second function after the processing of the message switches the network security layer from the initial state to a third state that restores the execution context before switching the network security layer back to the initial state.

In the initial state, the network security layer does not use any resources of the computing device. The return of the network security layer to its initial state without having to wait for the end of the processing of the message avoids inhibiting the computing device. The saving of the execution context makes it possible, at the end of the processing of the message, to return the network security layer to the context in which it was found before the operation began. Thus, the message securing operation is performed asynchronously.

A description of a particular embodiment of the invention follows, in reference to the figures, in which:

- Fig. 1 represents a secure network architecture;

- Fig. 2 represents a computing device for processing messages;

- Fig. 3 represents the essential stages of a secure operation layer, in the form of a machine with a finite number of states according to the prior art;

- Figs. 4 and 5 represent the essential stages of a secure operation layer in the form of a machine with a finite number of states according to the invention;

- Fig. 6 represents the essential stages of a hardware processing card driver in the form of a machine with a finite number of states for implementing the machine according to Figs. 3 and 4;

- Fig. 7 represents an architecture of save areas in memory;

- Fig. 8 presents a first step of a method for creating code for a network security layer;

- Fig. 9 presents a second step of the method for creating code for a network security layer;

- Fig. 10 presents a method for producing secure messages.

Referring to Fig. 1, a computing device 67 is physically linked to a first private network 69 and a computing device 68 is physically linked to a second private network 70. Messages can circulate in complete confidentiality through each of the private networks 69 and 70, insofar as no intrusion can be accomplished from outside these networks. However, if the device 67 sends an message to the device 68 using services of a public network 71, confidentiality is not assured without taking particular precautions. The public network 71 is for example the network known as the Internet, often represented in the form of a cloud in the literature. The public network 71 comprises several networks 72, 73, interconnected by means of computing devices such as a computing device 65, not controlled by the devices 67, 68.

The private network 69 is linked to the public network 71 by a computing device 66 and the private network 70 is linked to the public network 71 by a computing device 1. The computing devices 1 and 66 are called gateways in the rest of the description. Each computing device 1, 65, 66, 67, 68 traditionally includes a network layer using a communication protocol such as the known protocol IP, surmounted by a transport layer using a protocol such as the known protocol TCP, UDP or the like, in turn surmounted by an application layer such as http, ftp or the like, which send and receive messages. If a message passes through the TCP layer, then the IP layer in the device 67 and passes through the IP layer, then the TCP layer in the device 68, the routing of the message through the public network 71 normally stays within the IP layers of the devices 66, 65, 1.

However, the device 65 can facilitate an alien intrusion into the networks 72, 73, with the danger of intercepting the message in order to read, modify or even generate a message, by passing itself off as the device 67. One solution consists of encrypting and/or signing the message in the IP layer of the gateway 66, as it leaves the interconnecting network 72, then decrypting the message in the IP layer of the gateway 1, as it enters the interconnecting network 73. A solution known as Ipsec thus makes it possible to create a tunnel 74, which passes through the public network 71 in such a way as to create a virtual private network usable by the devices 67 and 68.

Referring to Fig. 2, a computing device 1 comprises a memory 2, one or more network access cards 3 and one or more cryptographic cards 4. The network access card 3 is designed to be connected to one or more physical links, not represented. The memory 2, of a known type such as a RAM, is designed to contain data and processing programs of the computing device 1. The network access card 3 is of a

known type, such as for example ethernet, for receiving and sending messages that flow through a computer network. The cryptographic card 4 is designed to encode and decode secure messages using dedicated hardware circuits that implement encryption algorithms of a known type, such as for example TripleDES. The dedicated hardware circuits, not represented, allow for a faster encoding and decoding operation than programs that are purely software. These circuits are not the subject of the present invention.

The memory 1 comprises data and programs of a user layer 5 and a kernel layer 6. The user layer 5 is a known type for executing applications, such as client or server applications on the Internet like http, www, telnet or others. The kernel layer 6 is designed to contain data structures and primitive functions of an operating system, such as for example the known UNIX operating system.

The kernel layer 6 comprises a network layer 7 and a driver 8. The network layer 7 is designed to execute network protocols, such as for example the IP protocol. The network layer 7 comprises a security layer 9 designed to execute secure communication protocols, such as for example Ipsec. The driver 8 is designed to control the cryptographic card 4, essentially at the request of the security layer 9.

Referring to Fig. 3, in an initial state 12, the network security layer 9 does not consume any resources of the system. Upon detection of a message to be secured, a transition 13, 14, 15, 16 switches the network security layer, respectively, into a state 17, 18, 19, 20, which calls a function F1, F2, F3, F4 for processing the message. At the return of the called function F1, F2, F3, F4, a transition 21, 22, 23, 24, indicating that the message has been processed, switches the network security layer 9 back to the initial state 12, thus freeing up the system resources required by the network security layer 9.

The transition 13 corresponds to the detection of a message M1 to be decrypted. The function F1 called is a function of the driver 8 that commands the cryptographic card 4 to decode the message. The cryptographic card is equipped with the algorithm and the keys required for decrypting the message. For example, in the case of the TripleDES algorithm, the cryptographic card uses the secret key to decode the message. When the cryptographic card 4 has finished decrypting the message, the driver 8 validates the transition 21, again making the message M1 available to the network security layer 9.

The transition 14 corresponds to the detection of a message M2 to be authenticated. The function F2 called is a function of the driver 8 that commands the

cryptographic card 4 to authenticate the message. The cryptographic card is equipped with the algorithm and the keys required for authenticating the message. For example, in the case of the HMAC-SHA1 algorithm, the cryptographic card uses the secret key to verify the signature of the gateway 66. When the cryptographic card 4 has finished authenticating the message, the driver 8 validates the transition 22, again making the message M2 available to the network security layer 9.

The transition 15 corresponds to the detection of a message M4 to be signed. The function F4 called is a function of the driver 8 that commands the cryptographic card 4 to sign the message. The cryptographic card is equipped with the algorithm and the keys required for signing the message. For example, in the case of the HMAC-SHA1 algorithm, the cryptographic card uses the secret key to generate its signature. When the cryptographic card 4 has finished signing the message, the driver 8 validates the transition 21, again making the message M4 available to the network security layer 9.

The transition 16 corresponds to the detection of a message M3 to be encrypted. The function F3 called is a function of the driver 8 that commands the cryptographic card 4 to encrypt the message. The cryptographic card is equipped with the algorithm and the keys required for encrypting the message. For example, in the case of the TripleDES algorithm, the cryptographic card uses the secret key to encode the message. When the cryptographic card 4 has finished encrypting the message, the driver 8 validates the transition 24, again making the message M3 available to the network security layer 9.

The disadvantage of the prior art described herein in reference to Fig. 3 is that the processing of the message has to be finished to allow the network security layer 9 to return to the initial state 12 and to free up the resources of the system, or to make it available for a subsequent processing of another or the same message. In essence, a message that is presented, for example the message M1 to be decrypted, can be presented as a message M2 to be authenticated after having been decrypted. All of the combinations are possible. The encryption and decryption operations are particularly long, and may even be performed by means of hardware circuits.

Referring to Fig. 4, in an initial state 12, the network security layer 9 does not consume any resources of the system. Upon detection of a message M1, M2, M4, M3 to which a security operation is to be applied, a transition 13, 14, 15, 16 switches the network security layer to a respective state 25,26, 27, 28, which activates a sequence F5, F6, F7, F8 for saving the current execution context CE. At the end of the sequence

F5, F6, F7, F8, a transition 29, 30, 31, 32 is validated by a pointer value PZS(M1), PZS(M2), PZS(M4), PZS(M3) in a save area resulting from the preceding state 25, 26, 27, 28.

The security operations - decryption operations downstream from the transition 13, authentication downstream from the transition 14, signature downstream from the transition 15, and encryption downstream from the transition 16 - are considered as non-limiting examples in reference to Figs. 3 and 4, comparatively to Fig. 3. The teaching of the invention is also valid for any other operation such as message digesting or message compression.

Each save sequence F5, F6, F7, F8 is specific to the operation to be performed for each type of message M1, M2, M4, M3. The sequence F5 F6, F7, F8 essentially consists of saving the current execution context CE in a storage area. The current execution context CE is constituted by local and global variables that are used by the network security layer 9 to process the message, such as the security characteristics of the message, and the protocols and keys to be used. The start of the storage area is marked by a pointer PZS(M1), PZS(M2), PZS(M4), PZS(M3) so that the execution context CE linked to the processing of the message M1, M2, M4, M3 can subsequently be restored.

When the sequence F5 has finished saving the execution context CE, the transition 29 switches the network security layer 9 to a state 33, which performs a call to a function F9 executed by the driver 8 in order to command the card 4 to decrypt the message M1. The function F9 passes as parameters a so-called function return address @F13, a so-called correlation variable VC1, and the value of the pointer PZS(M1).

A transition 37 is validated by an acknowledgement of the function F9, returned by the driver 8. The transition 37 switches the network security layer 9 back to its initial state 12.

When the sequence F6 has finished saving the execution context CE, the transition 30 switches the network security layer 9 to a state 34 that performs a call to a function F10 executed by the driver 8 in order to command the card 4 to authenticate the message M2. The function F10 passes as parameters a so-called function return address @F14, a so-called correlation variable VC2, and the value of the pointer PZS(M2).

A transition 38 is validated by an acknowledgement of the function F10 returned by the driver 8. The transition 38 switches the network security layer 9 back to its initial state 12.

When the sequence F7 has finished saving the execution context CE, the transition 31 switches the network security layer 9 to a state 35 that performs a call to a function F11, executed by the driver 8 in order to command the card 4 to sign the message M4. The function F11 passes as parameters a so-called function return address @F15, a so-called correlation variable VC4, and the value of the pointer PZS(M4).

A transition 39 is validated by an acknowledgement of the function F11, returned by the driver 8. The transition 39 switches the network security layer 9 back to its initial state 12.

When the sequence F8 has finished saving the execution context CE, the transition 32 switches the network security layer 9 to a state 36 that performs a call to a function F12, executed by the driver 8 in order to command the card 4 to sign the message M3. The function F12 passes as parameters a so-called function return address @F16, a so-called correlation variable VC3, and the value of the pointer PZS(M3).

A transition 40 is validated by an acknowledgement of the function F12, returned by the driver 8. The transition 40 switches the network security layer 9 back to its initial state 12.

Fig. 6 presents the states and transitions of the cryptography card driver 8 that are specifically adapted for interfacing with the states and transitions of the network security layer 9 according to the invention, in reference to Figs. 3 and 4. Other states of the driver, applicable to the control of the card 4, are not described herein, as those other states are beyond the scope of the present invention. The states described are the ones that correspond to the encryption and decryption operations. The resulting teaching is applicable to authentication, signature, or to any other securing operation such as message digesting by means of the hardware card 4.

In an initial state 41, the driver 8 does not use any resources of the system. A transition 42 is activated by a call of the function F9, performed in the state 33 of the network security layer 9. A transition 43 is activated by a call of the function F12, performed in the state 36 of the network security layer 9.

The transition 42 switches the driver 8 to a state 44. In the state 44, the driver 8 immediately sends an acknowledgement Ack(F9) that validates the transition 37 and

activates the card 4 in order to perform a hardware decryption operation on the message M1. The card 4 then processes the message M1. As soon as the card 4 is activated, a transition 46 switches the driver back to the initial state 41, which makes it available to handle other requests for processing by the network security layer 9.

When the card 4 has finished decrypting the message M1, a transition 48 switches the driver to a state 50. In the state 50, the driver performs a jump to the function return address @F13 by communicating the pointer PZS(M1) given previously in the state 33 of the network security layer. The driver also enters into the correlation variable VC1 the coordinates at which the message M1 decrypted by the card 4 is available. The driver then returns to its initial state 41.

The transition 43 switches the driver 8 to a state 45. In the state 45, the driver 8 immediately sends an acknowledgement Ack(F12), which validates the transition 40 and activates the card 4 in order to perform a hardware operation for encrypting the message M3. The card 4 then processes the message M3. As soon as the card 4 is activated, a transition 47 switches the driver back to the initial state 41, which makes it available to handle other requests for processing by the network security layer 9.

When the card 4 has finished encrypting the message M3, a transition 49 switches the driver to a state 51. In the state 51, the driver performs a jump to the function return address @F16 by communicating the pointer PZS(M3) given previously in the state 36 of the network security layer. The driver also enters into the correlation variable VC3 the coordinates at which the message M3 encrypted by the card 4 is available. The driver then returns to its initial state 41.

Referring to Fig. 5, a transition 52 switches the network security layer from the initial state 12 to a state 56, a transition 53 switches the network security layer from the initial state 12 to a state 57, a transition 54 switches the network security layer from the initial state 12 to a state 58, and a transition 55 switches the network security layer from the initial state 12 to a state 59.

The transition 52 is validated by the jump to the address @F13 and the communication of the pointer PZS(M1) performed in the state 50. In the state 56, the network security layer 9 restores the execution context saved in the storage area pointed to by PZS(M1). The network security layer 9 thus returns to the configuration it had when it was in the state 25 for the message M1 when the message M1 was not decrypted. However, now that the message is decrypted, the correlation variable VC1 immediately validates a transition 60 that returns the network security layer to its initial state 12. The correlation variable VC1 makes the message M1 available to the

network security layer 9 in order to make other functions of the network layer available or to present the processed message M1 as a message of the M2, M3, M4 type for another operation. In order to make the message M1 available to the network security layer 9, the value of the correlation variable VC1 is for example a value that makes it possible to resume execution at an appropriate place.

The transition 55 is validated by the jump to the address @F16 and the communication of the pointer PZS(M3), performed in the state 51. In the state 59, the network security layer 9 restores the execution context saved in the storage area pointed to by PZS(M3). The network security layer 9 thus returns to the configuration it had when it was in the state 28 for the message M3 when the message M3 was not encrypted. However, now that the message is encrypted, the correlation variable VC3 immediately validates a transition 64, which returns the network security layer to its initial state 12. The correlation variable VC3 makes the message M3 available to the network security layer 9 in order to make other functions of the network layer 7 available or to present the processed message M3 as a message of the M2, M1, M4 type for another operation.

Likewise, the transition 53 is validated by the jump to the address @F14 and the communication of the pointer PZS(M2) performed in a non-represented state of the driver 8. In the state 57, the network security layer 9 restores the execution context saved in the storage area pointed to by PZS(M2). The network security layer 9 thus returns to the configuration it had when it was in the state 26 for the message M2 when the message M2 was not authenticated. However, now that the message is authenticated, the correlation variable VC2 immediately validates a transition 62, which returns the network security layer to its initial state 12. The correlation variable VC2 makes the message M2 available to the network security layer 9 in order to make other functions of the network layer 7 available or to present the processed message M2 as a message of the M1, M3, M4 type for another operation.

Likewise, the transition 54 is validated by the type to the address @F15 and the communication of the pointer PZS(M4) performed in a non-represented state of the driver 8. In the state 58, the network security layer 9 restores the execution context saved in the storage area pointed to by PZS(M4). The network security layer 9 thus returns to the configuration it had when it was in the state 27 for the message M4 when the message M2 was not signed. However, now that the message is signed, the correlation variable VC4 immediately validates a transition 63, which returns the network security layer to its initial state 12. The correlation variable VC4 makes the

message M4 available to the network security layer 9 in order to make other functions of the network layer 7 available or to present the processed message M4 as a message of the M1, M3, M2 type for another operation.

In Fig. 2 , we take a path 10 of an encrypted message M1 from the network card 3 to the cryptographic card 4, followed by a path 11 of the decrypted message M1 from the card 4 to the memory 2 for its presentation, for example, to the user layer 5.

When the message M1 coming from the card 3 is transmitted to the memory 2 on the ascending branch of the path 10, its presentation to the network security layer 9 validates the transition 13. The network security layer 9 is only in the state 25 for a short time, since the saving of the execution context is a relatively fast operation. After the state 25, the network security layer 9 is only in the state 33 for a short time, since the state 44 of the driver 8 sends the acknowledgement Ack(F9) immediately after the call of the function F9 without waiting until the message M1 is decrypted. The network security layer 9 then quickly returns to its initial state 12. This prevents the system from being inhibited during the operation for decrypting the message M1, since this operation is handled by the card 4 asynchronously. Furthermore, it has the advantage of quickly making the network security layer available again for the presentation of another message to be processed.

When the message M1 is stored in decrypted form by the card 4 in memory 2 on a first ascending branch of the path 11, the state 50 of the driver 8 validates the transition 52 of the network security layer 9. The network security layer 9 is only in the resulting state 56 for a short time, since the restoration of the execution context CE is a relatively fast operation. After the restoration of the context CE, the transition 21 quickly returns the network security layer 9 to the initial state 12, since the correlation value VC1 immediately makes the message M1 in decrypted form available to the network security layer 9 so that it can be retransmitted, in the case of Fig. 2, to the user layer 5 via a second ascending branch of the path 11. Thus, the decryption time of the message M1 is totally transparent for the network security layer 9, activated only a short time after the presentation of the message M1 to be decrypted, then reactivated only a short time after the presentation of the decrypted message M1.The paths 10 and 11 of Fig. 2 are symbolic, and simply illustrate the advantages of the invention. One skilled in the art would also know that one or more layers could separate the network layer 7 from the user layer 5, such as a transport layer of the known TCP type, not represented in order not to unnecessarily

overcomplicate Fig. 2. Furthermore, the path 11 could also be redirected to the card 3 by the network layer 7 or back to the card 4 for a subsequent operation.

Since the network layer 6 is not inhibited while waiting for the processing of a message to end, it is advantageous to provide for other messages that are presented to the network security layer 9 be handled while a first message has not yet finished being processed.

Referring to Fig. 7, while the message M1 is handled by the card 4 in order to be decrypted, the pointer PZS(M1) has the value of a word 56 that contains a start address of an area 52 of the memory 2. The area 52 contains the execution context CE that the network security layer had when it was in the state 25 for the message M1. A word 55 is designed to contain an address that follows a last address of the area 52. Thus, the word 55 defines a pointer PZL to an available area in a subsequent executing context save area 53.

When another message M'1 is presented to the network security layer 7, the value of the word 55 is transferred into a word 57 so as to define a new pointer PZS(M'1) at the beginning of the area 53 in which the execution context CE is saved when the network security layer is in the state 25 for the message M'1. The word 55 therefore contains an address that follows a last address of the area 53. The word 55 defines a pointer PZL to an available area in a subsequent execution context save area 54, available for the execution context CE linked to a new message M"1. This process is repeated for any new message in order to chain the saving of execution contexts CE.

After a restoration of an execution context CE in the state 56 of the network security layer, the start address of the released save area is taken to be the address that follows the last save area occupied, using a standard chaining mechanism.

It is possible to use a data structure similar to that just described, distinct for each of the states 25, 26, 27, 28 of the network security layer, or common to all the states 25, 26, 27, 28, in which case the words 56, 57 can contain a PZS(M1), PZS(M2), PZS(M3), PZS(M4) for any one of these states.

The network security layer can be programmed in various ways in order to implement the states described above. One method for creating code for the network security layer 9 from a standard network security layer such as for example the Ipsec layer of LINUX, essentially comprises two steps.

The first step is explained in reference to Fig. 8. In the kernel layer 6 of the computing device 1, a first code sequence 75 is designed to be activated by the

presentation of a message M1, M2, M3 or M4 to which a securing operation - encryption, authentication, encryption or signature - is to be applied. In the standard network security layer, the code sequence 75 is constituted by several lines of standard code which are not the subject of the present invention. At this stage, there is only a line 76 and a last line of the sequence 75, indicated by an End indicator. The line 76 contains a call to the standard securing operation function, for example the first function F1 if the code sequence 75 is the one activated by the presentation of the message M1.

The first code sequence 75 is modified by inserting, ahead of line 75, a second code sequence 77. The code sequence 77 begins with one or more lines F5(CE), which save the current execution context CE when the first sequence is activated, i.e. essentially the values of the local and global variables used in the code sequence 75. The save code therefore consists in the writing of values of these variables into an area of the memory 2, indicated by the pointer PZS(M1).

After the lines F5(CE), the sequence 77 contains the code for calling a second security function, for example the function F9(@F13, VC1, PZS(M1)) in the case described here. The second function is designed to be executed by the driver 8. The parameters passed are essentially a function address @F13 and the pointer PZS to the save area.

The code sequence 77 ends in a jump to the last line of the code sequence 75 of the "Goto End" type.

The second step is explained in reference to Fig. 9. The first code sequence 75 is copied so as to generate a third code sequence 78, taken to be the code of the function F13 whose address @F13 is indicated by a pointer 81. A fourth code sequence 80 is inserted after the line 76 of the sequence 78. The sequence 80 is indicated by a label and contains start of the sequence 78. The line 79 contains a jump instruction "Goto Label" to the code sequence 80.

The network security layer (9) obtained by means of the method described above, is faster than the original standard network security layer. In essence, in the standard security layer, the execution of the unmodified sequence 75 takes place in the following way. The standard code instructions that precede the line 76 are executed. The line 76 executes a call to the standard processing function F1. The standard code instructions that follow the line 76 are executed after the return of the function F1, which indicates the end of the processing of the message. But a

cryptographic operation is intrinsically long. This has the effect of slowing down the wait for the execution of the last line "End" of the unmodified sequence 75.

In the network security layer obtained by means of the method, the execution of the modified sequence 75 takes place in the following way. The standard code instructions that precede the line 76 and the sequence 77 are executed. The line 76 and the subsequent lines of the sequence 75 are never executed because of the first jump to the last line of the sequence 75. The first jump is performed quickly because the function F9 immediately sends an acknowledgement before the message is finished being processed. When the processing of the message is finished, the driver 8 triggers an execution of the code sequence 78 by means of the address @F13. The code line 76 and the code lines of the sequence 78 that precede it are never executed because of the jump at the start of the sequence 78 to the sequence 80, which allows the execution of the subsequent lines of code, thus masking the processing time of the message.

The computing device just described makes it possible to implement a method for maintaining a secure message from another message.

Referring to Fig. 10, upon presentation of said other message to the network security layer, in a first step 82, the current execution context is saved. This step is performed in one of the states 25, 26, 27, 28 of the layer 9. In a second step 83, a request for a securing operation is sent from the layer 9, in one of the states 33, 34, 35, 36, to an element outside the layer 9, so that the layer 9 is returned to its initial state, which does not use any resources of the device. Steps 82 and 83 are implemented by means of the sequence 77. After the external element has processed said other message, the saved context is restored in a step 84 in order to produce the secure message.

This method has the advantage of being able to produce secure messages in large numbers, since step 84 can be activated after several successive activations of the steps 82, 83 for different messages

CLAIMS

1. Computing device (1) comprising a memory (2) and a network security layer (9) for applying a securing operation upon presentation of a message (M1) in the memory (2), characterized in that:

- the presentation of the message (M1) switches the network security layer (9) from an initial state (12) to a first state (25) that saves an execution context (CE) in an area (52) of the memory (2);

- the saving of the execution context (CE) switches the network security layer from the first state (25) to a second state (33) that calls a first function (F9) for processing the message (M1), passing as parameters of said first function (F9), at least an address (@F13) of a second function (F13) and a pointer PZS(M1) to the area (52) of the memory (2);

- an acknowledgement of the first function (F9) before the processing of the message (M1) immediately switches the network security layer back to the initial state (12);

- a jump to the address (@F13) of the second function switches the network security layer (9) from the initial state (12) to a third state (56) that restores the execution context (CE) before switching the network security layer (9) back to the initial state.

2. Computing device (1) according to claim 1, characterized in that several pointers PZS(M1), PZS(M'1) are chained so that they can be restored at the time of the jump to said address (@F13).

3. Computing device (1) according to claim 1 or 2, characterized in that the call of the first function (F9) passes as a parameter a correlation variable (VC1) restored at the time of the jump to the address (@F13).

4. Method for creating code for a fast network security layer (9) from the code of a standard network security layer in a kernel layer (6) of a computing device (1), characterized in that it comprises:

- a first step for modifying, in the code of said standard layer, a first code sequence designed to be activated by the presentation of a message to which a securing

operation is to be applied, by inserting into the first sequence, before calling a first securing function (F1), a second code sequence that:

- begins by saving a current execution context (CE) when the first sequence is executed,
- makes a call to a second securing function (F9),
-ends with a first jump to the end of the first code sequence;

- a second step for generating a third code sequence of a third function (F13) by copying said first modified code sequence, then inserting said third code sequence into it:

- after the call to the first function (F1), a fourth code sequence for restoring the saved execution context (CE),
- at the start of the third sequence, a second jump to said fourth code sequence.

5. Method for obtaining a secure message from another message, by means of a computing device (1) comprising a network security layer (9) to which said other message is presented, characterized in that it comprises:

- a first step for saving an execution context of the network security layer after the presentation of said other message;
- a second step in which the network security layer sends a request for a securing operation to an element outside the network security layer such that said external element immediately acknowledges this request in order to place the network security layer in an initial state that does not use any resources of the computing device (1);
- a third step in which said external element activates a restoration of the saved execution context in the network security layer by presenting the message secured by the securing operation that results from said request.

ABSTRACT

COMPUTER DEVICE FOR MAKING SECURE MESSAGES AT A NETWORK LAYER

The computing device (1) comprising a memory (2) and a network security layer (9) for applying a securing operation upon presentation of a message (M1) in the memory (2) is characterized in that:

- the presentation of the message (M1) switches the network security layer (9) from an initial state (12) to a first state (25) that saves an execution context (CE) in an area (52) of the memory (2);

- the saving of the execution context (CE) switches the network security layer from the first state (25) to a second state (33) that calls a first function (F9) for processing the message (M1), passing as parameters of said first function (F9) at least an address (@F13) of said function (F13) and a pointer PZS(M1) to the area (52) of the memory (2);

- an acknowledgement of the first function (F9), before the processing of the message (M1), immediately switches the network security layer back to the initial state (12);

- a jump to the address (@F13) of a second function switches the network security layer (9) from the initial state (12) to a third state (56) that restores the execution context (CE) before switching the network security layer (9) back to the initial state.

Fig. 2

#9150912

179

Fig.1

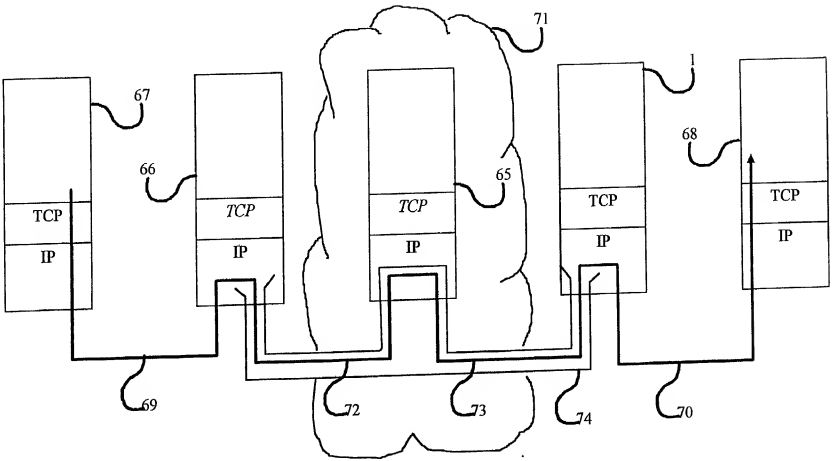
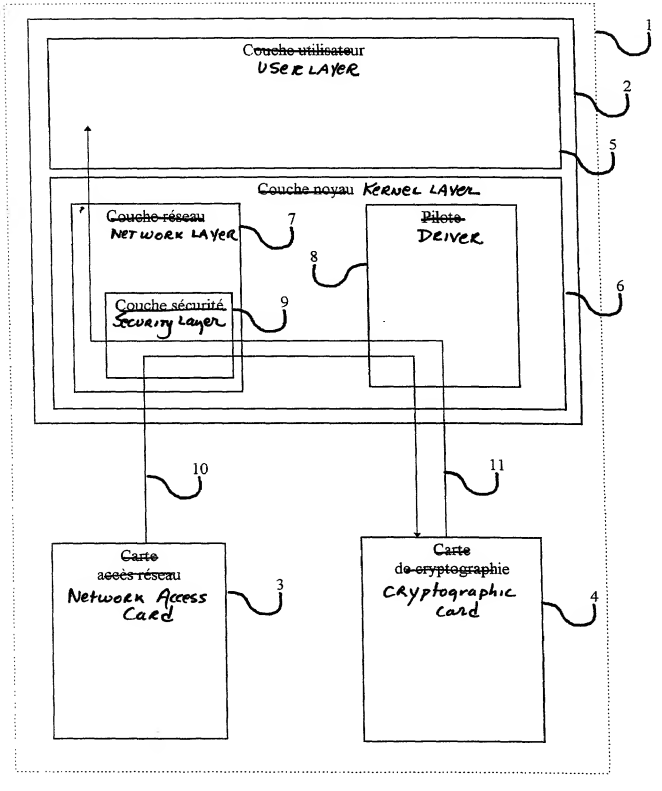


Fig.2



~~379~~

Fig.3

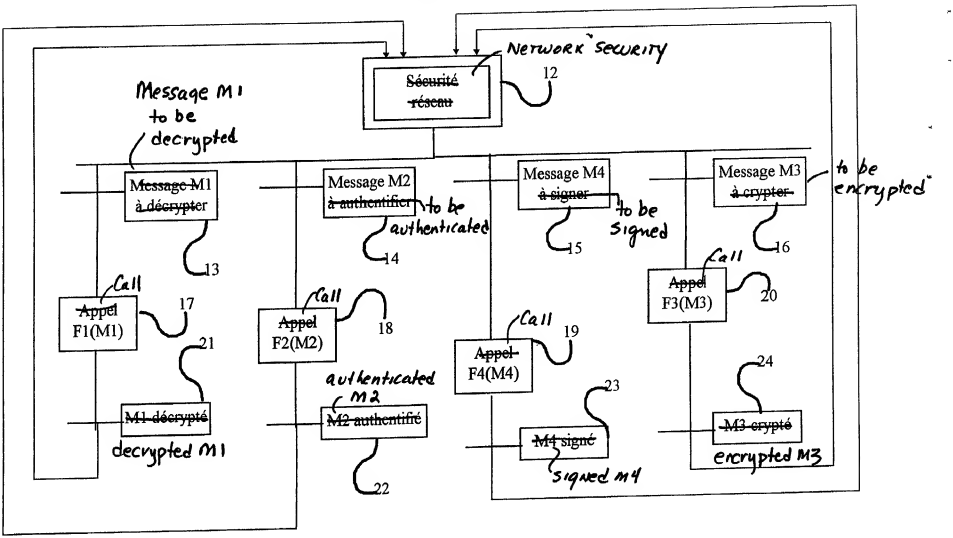


Fig.4

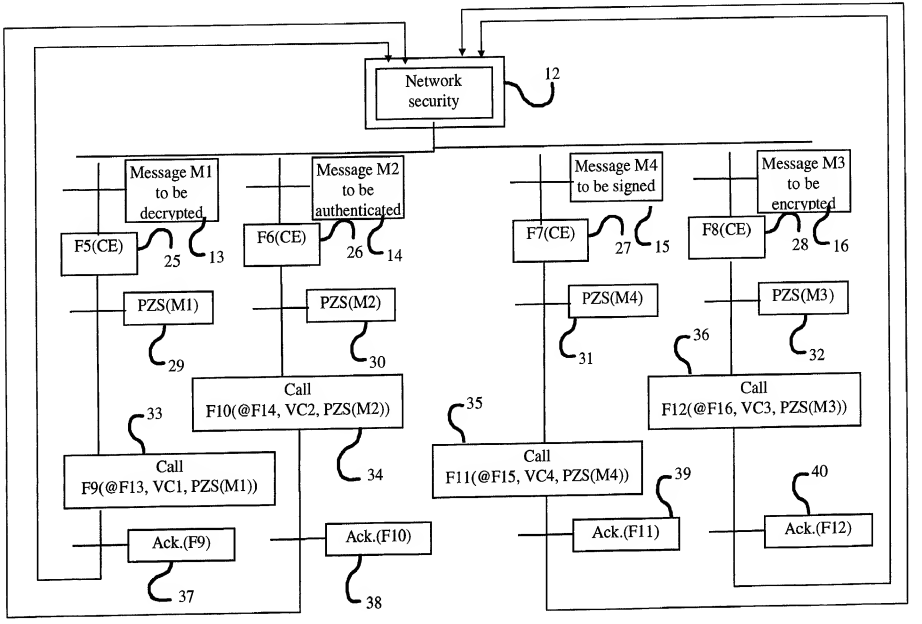


Fig.5

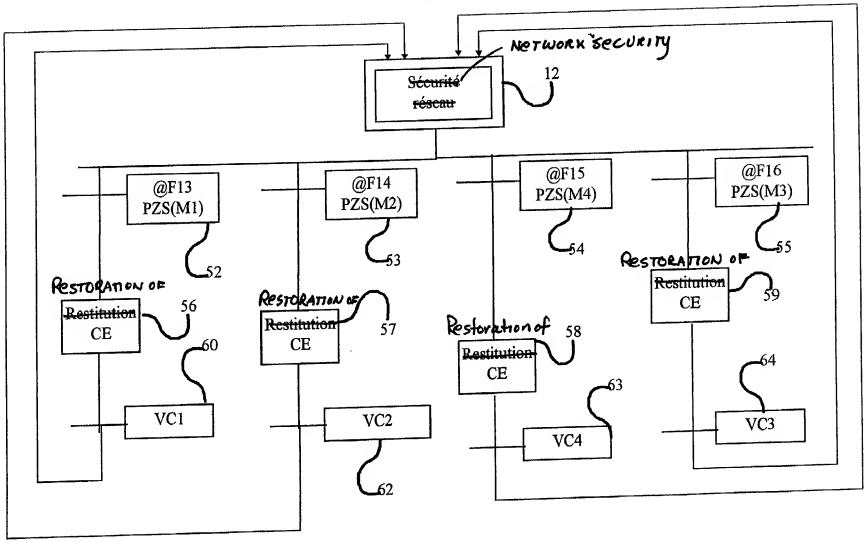


Fig.6

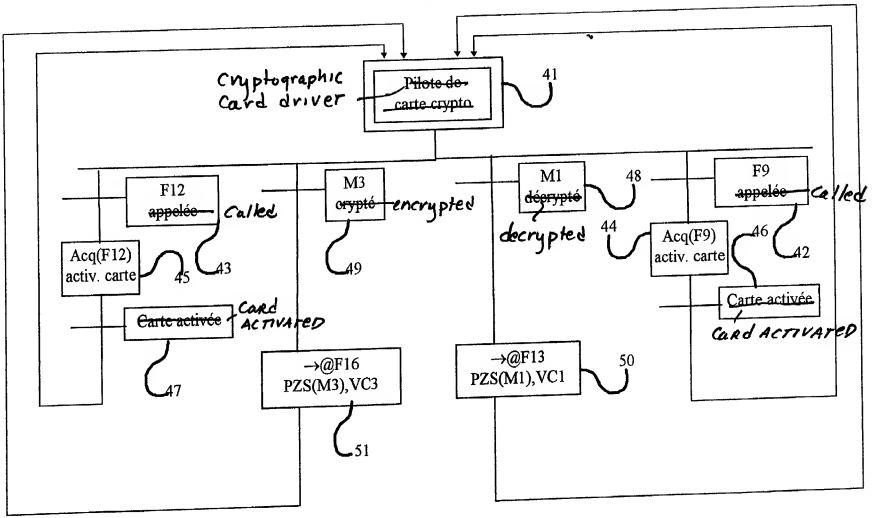
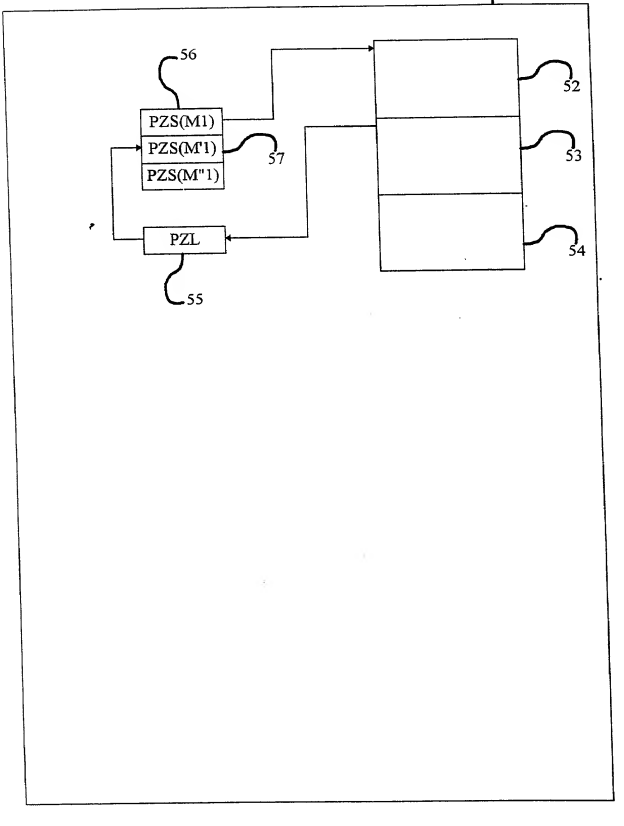


Fig.7



8/9

Fig.8

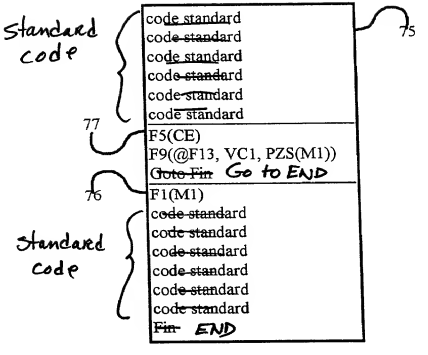


Fig.9

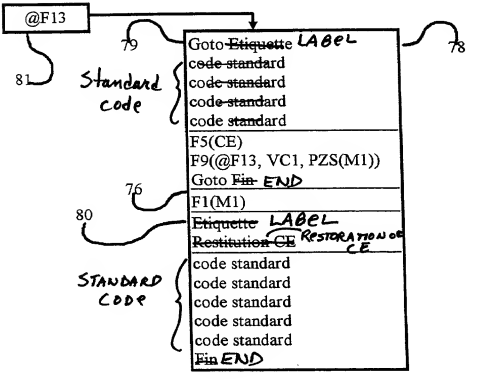
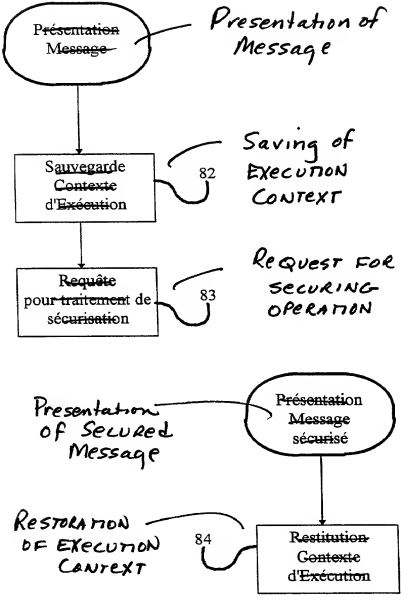


Fig. 10



Declaration and Power of Attorney For Patent Application
Declaration Pour Demandes de Brevets Avec Pouvoirs

French Language Declaration

En tant qu' inventeur nomme ci-après, Je déclare par le présent acte que:

As a below named inventor, I hereby declare that:

Mon nom, mon domicile, mon adresse postale, ma nationalité sont ceux qui figurent ci-après,

My residence, post office address and citizenship are as stated below next to my name,

Je déclare que je crois être l'inventeur original, premier et unique (si un seul nom figure sur le présent acte) ou un des co-inventeurs, originaux et premiers (si plusieurs noms figurent sur le present acte) du sujet revendiqué et pour lequel un brevet est demande sur la base de l'invention intitulée:

Dispositif informatique pour sécuriser des messages au niveau d'une couche réseau

I believe I am the original, first and sole inventor (if only one name is listed below) or an original, first and joint inventor (if plural names are listed below) of the subject matter which is claimed and for which a patent is sought on the invention entitled

dont la description (cocher la case correspondante)

the specification of which (check one)

☐ est annexée au présent acte.

☐ a été déposée _____

Numéro de série de la demande _____

et modifiée le _____ (si approprié)

☐ is attached hereto.

☐ was filed on _____ as Application Serial No. _____

and was amended on _____ (if applicable)

Je déclare par le présent acte avoir examiné et compris le contenu de la description identifiée ci-dessus, revendications y compris, et le cas échéant telle que modifiée par l'amendment cité plus haut.

I hereby state that I have reviewed and understand the contents of the above identified specification, including the claims, as amended by any amendment referred to above.

Je reconnais le devoir de divulguer l'information qui est en rapport avec l'examen de cette demande selon Titre 37 du Code des Reglements Fédéraux §1.56(a).

I acknowledge the duty to disclose information which is material to the examination of this application in accordance with Title 37, Code of Federal Regulations, §1.56(a).

French Language Declaration

Je revendique par le présent acte le bénéfice de priorité étrangère selon Titre 35, du Code des Etats-Unis, §119 de toute demande de brevet ou d'attestation d'inventeur énumérée ci-après, et j'ai identifié également ci-après toute demande étrangère de brevet ou d'attestation d'inventeur ayant une date de dépôt antérieure à celle de la demande pour laquelle la priorité est revendiquée.

I hereby claim foreign priority benefits under Title 35, United States Code, §119 of any foreign application(s) for patent or inventor's certificate listed below and have also identified below any foreign application for patent or inventor's certificate having a filing date before that of the application on which priority is claimed:

Prior foreign applications

Demande(s) de brevet anterieure(s) dans un autre pays:

(Number) (Numéro)	(Country) (Pays)	(Day/Month/Year Filed) (Jour/Mois/Année de dépôt)	Priority claimed Droit de priorité revendiqué Yes Oui	No Non
99 14755	FRANCE	23.11.99	X	

Je revendique par le présent acte, le bénéfice selon Titre 35 du Code des Etats-Unis, §120 de toute(s) demande(s) américaines énumérée(s) ci-après et, dans la mesure où le sujet de chacune des revendications de cette demande n'est pas divulgué dans la demande américaine antérieure, de la façon définie par le premier paragraphe de Titre 35 du Code des Etats-Unis, §112, je reconnais le devoir de divulguer l'information pertinente selon Titre 37 du Code des Réglements Fédéraux, §1.56(a), toute information qui se présente entre la date de dépôt de la demande antérieure et la date de dépôt de la demande, soit nationale, soit internationale PCT.

I hereby claim the benefit under Title 35, United States Code, §120 of any United States application(s) listed below and, insofar as the subject matter of each of the claims of this application is not disclosed in the prior United States application in the manner provided by the first paragraph of Title 35, United States Code, §112, I acknowledge the duty to disclose material information as defined in Title 37, Code of Federal Regulations, §1.56(a) which occurred between the filing date of the prior application and the national or PCT international filing date of this application:

(Application Serial No.) (No. de Demande)	(Filing Date) (Date de Dépôt)	(Etat) (brevetée, pendante, abandonné)	(Status) (patented, pending, abandoned)

Je déclare par le présent acte que toutes mes déclarations, à ma connaissance, sont vraies et que toutes les déclarations faites à partir de renseignements ou de suppositions, sont tenues pour être vraies; de plus, toutes ces declarations ont été faites en sachant que de fausses déclarations volontaires u autres actes de même nature sont sanctionées par une amende ou un emprisonnement, ou les deux, selon la Section 1001, du Titre 18 de Code des Etats-Unis et que de selles déclarations délibérément fausses peuvent compromettre la validité de la demande ou du brevet délivré.

I hereby declare that all statements made herein of my own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

French Language Declaration

POUVOIR: En tant qu'inventeur, je désigne l'(les) avocat(s) et/ou l'(les) agent(s) suivant(s) pour poursuivre la procédure de cette demande et traiter toute affaire la concernant supris du Bureau des Brevets et de Marques:

Harold L. Stowell, Reg. 17,233
Edward J. Kondracki, Reg. 20,604
Dennis P. Clarke, Reg. 22,549
William L. Feeney, Reg. 29,918
John C. Kerins, Reg. 32,421

POWER OF ATTORNEY: As a named inventor, I hereby appoint the following attorney(s) and/or agent(s) to prosecute this application and transact all business in the Patent and Trademark Office connected therewith. (*list name and registration number*)

Harold L. Stowell, Reg. 17,233
Edward J. Kondracki, Reg. 20,604
Dennis P. Clarke, Reg. 22,549
William L. Feeney, Reg. 29,918
John C. Kerins, Reg. 32,421

Adresser toure correspondance à:
Edward J. Kondracki, Esq.
KERKAM, STOWELL, KONDRACKI & CLARKE, P.C.
5203 Leesburg Pike, Suite 600
Falls Church, VA 22041

Send Correspondence to:
Edward J. Kondracki, Esq.
KERKAM, STOWELL, KONDRACKI & CLARKE, P.C.
5203 Leesburg Pike, Suite 600
Falls Church, VA 22041

Adresser toute communication téléphonique à: (*Nom*) (*Numéro de téléphone*)
Edward J. Kondracki, Esq.
(703) 998-3302

Direct Telephone Calls to: (*name and telephone number*)
Edward J. Kondracki, Esq.
(703) 998-3302

Nom complet du seul ou premier inventeur **Cunchon François**	Full name of sole or first inventor
Signature de l'inventeur — Date 7 décembre 1999	Inventor's signature — Date
Domicile 5, rue Claude Nicolas Ledoux, 78114 Magny les Hameaux, France FRX	Residence
Nationalité Française	Citizenship
Adresse Postale 5, rue Claude Nicolas Ledoux, 78114 Magny les Hameaux, France	Post Office Address
Nom complet du second co-inventeur, le cas echeant **Martin René**	Full name of second joint inventor, if any
Signature du Inventeur — Date 7 Decembre 1999	Second Inventor's signature — Date
Domicile 32, rue de Gometz, 91440 Bures sur Yvette France FRX	Residence
Nationalité Française	Citizenship
Adresse Postale 32, rue de Gometz, 91440 Bures sur Yvette France	Post Office Address

(Fournir les mêmes renseignements et la signature de tout co-inventeur supplémentaire.)

(Supply similar information and signature for third and subsequent joint inventors.)

(3) over

French Language Declaration

Nom complet du troisième inventeur Tran Minh Lap	Full name of third joint inventor, if any
Signature de l'inventeur Date 7 Dec 2001	Inventor's signature Date
Domicile 18, rue Paul Eluard, 95360 Montmagny, France	Residence FRX
Nationalité Française	Citizenship
Adresse Postale 18, rue Paul Eluard, 95360 Montmagny, France	Post Office Address
Nom complet du quatrième inventeur	Full name of fourth joint inventor, if any
Signature de l'inventeur Date	Inventor's signature Date
Domicile	Residence
Nationalité	Citizenship
Adresse Postale	Post Office Address

T2147-907330-US3876/JMD(PCT)

IN THE UNITED STATES DESIGNATED/ELECTED OFFICE (D.O./E.O./US)

Applicant:	Francois CUNCHON et al.
International Application No.:	PCT/FR00/03230
International Filing Date:	21 November 2000
U.S. Serial No.:	To be Assigned
U.S. Filing Date:	July 23, 2001
For:	**COMPUTER DEVICE FOR MAKING SECURE MESSAGES AT A NETWORK LAYER**

McLean, Virginia

CHANGE OF ADDRESS

Honorable Commissioner of Patents and Trademarks
Washington, D.C. 20231

Sir:

Effective immediately, please note our new correspondence address and telephone/fax numbers as follows:

Miles & Stockbridge P.C.
1751 Pinnacle Drive
Suite 500
McLean, VA 22102-3833
Telephone: 703-903-9000
Fax: 703-610-8686

Respectfully submitted,

MILES & STOCKBRIDGE P.C.

Date: July 23, 2001

By: _____
Edward J. Kondracki
Registration No. 20,604

1751 Pinnacle Drive – Suite 500
McLean, VA 22102-3833
Tel.: 703/903-9000

TYSC01:915066v01T2147-907330\07\2001